

Appl. No. 10/798,079
Filed. March 11, 2004
Reply to Office action of September 11, 2006

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (previously presented): A method for detecting attempted intrusions in a database application, the method comprising:

monitoring for an SQL statement, said SQL statement executable in said database application and intended to exploit a vulnerability;
actuating said SQL statement to discover an atomic SQL command;
analyzing said atomic SQL command against a pre-defined set of detection rules.

Claim 2 (previously presented): The method according to claim 1, wherein said vulnerability is a buffer overflow in a SQL procedure.

Claim 3 (previously presented): The method according to claim 1, wherein said vulnerability is a buffer overflow in a call from SQL to an operating system function.

Claim 4 (previously presented): The method according to claim 1, wherein said vulnerability is an attempt to escalate privileges of a user in said database application.

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

Claim 5 (previously presented): The method according to claim 1, wherein said vulnerability is an attempt to escalate privileges within an operating system.

Claim 6 (previously presented): The method according to claim 1, wherein said vulnerability is an attempt to insert an invasive SQL statement into a parameter of stored procedures.

Claim 7 (previously presented): A method for detecting an anomalous command in a database application, the method comprising:

actuating said database application in order to discover a form of a set of authorized SQL statements and commands and to discover appropriate parameters for said statements and commands;

generating a rule set of said discovered form of said authorized SQL statements;

monitoring for SQL statements executable in said database application which do not match said generated rule set of forms of authorized SQL statements.

Claim 8 (previously presented): The method according to claim 7, wherein said anomalous command is a SELECT statement.

Claim 9 (previously presented): The method according to claim 7, wherein said anomalous command is an UPDATE statement.

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

Claim 10 (previously presented): The method according to claim 7, wherein said anomalous command is an INSERT statement.

Claim 11 (previously presented): The method according to claim 7, wherein said anomalous command is a DELETE statement.

Claim 12 (previously presented): The method according to claim 7, wherein said anomalous command is a call to a stored procedure.

Claim 13 (previously presented): The method according to claim 7, wherein said anomalous command is a batch script.

Claim 14 (previously presented): A method for detecting attempts to access a database application from invalid sources, the method comprising:

actuating said database application in order to discover a normal set of authorized SQL sources;

generating a rule set of characteristics of connecting at least one of said normal set of SQL sources;

monitoring for SQL statements executable in said database application which do not match said generated rule set of valid forms for authorized SQL statements.

Claim 15 (previously presented): The method according to claim 14, wherein a characteristic of said rule set is based on a location of an SQL source.

Appl. No. 10/798,079
Filed. March 11, 2004
Reply to Office action of September 11, 2006

Claim 16 (previously presented): The method according to claim 14, wherein a characteristic of said rule set is based on a network address of an SQL source.

Claim 17 (previously presented): The method according to claim 14, wherein a characteristic of said rule set is based on a host name of an SQL source.

Claim 18 (previously presented): The method according to claim 14, wherein a characteristic of said rule set is based on a domain name of an SQL source.

Claim 19 (previously presented): The method according to claim 14, wherein a characteristic of said rule set is based on a time of activity of an SQL source.

Claim 20 (previously presented): The method according to claim 14, wherein a characteristic of said rule set is based on an application name of an SQL source.

Claim 21 (previously presented): The method according to claim 14, wherein a characteristic of said rule set is based on a behavior of an SQL source.

Claim 22 (previously presented): A method for detecting unauthorized activity in a database application, the method comprising:

monitoring for SQL statements executable in said database application and intended to perform activities not authorized by an SQL source;

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

actuating each discrete database event;

analyzing each event against a pre-defined set of detection rules.

Claim 23 (previously presented): The method according to claim 22, wherein said unauthorized activity is accessing data for which said SQL source has not been granted privileges.

Claim 24 (previously presented): The method according to claim 22, wherein said unauthorized activity is accessing data not using an authorized method.

Claim 25 (previously presented): The method according to claim 22, wherein said unauthorized activity is accessing data in a data dictionary not using an authorized method.

Claim 26 (previously presented): The method according to claim 22, wherein said unauthorized activity is interfering with auditing settings.

Claim 27 (previously presented): The method according to claim 22, wherein said unauthorized activity is interfering with audit records.

Claim 28 (previously presented): The method according to claim 22, wherein said unauthorized activity is modifying configuration settings.

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

Claim 29 (previously presented): The method according to claim 22, wherein said unauthorized activity is modifying security settings.

Claim 30 (previously presented): The method according to claim 22, wherein said unauthorized activity is a use of an unauthorized tool to attempt to access said database application.

Claim 31 (previously presented): A method for detecting activity designed to breach security of a database application, the method comprising:

monitoring for discrete events executable in said database application and intended to breach a security mechanism associated with said database application;
actuating each discrete database event;
analyzing said database events against a pre-defined set of detection rules.

Claim 32 (previously presented): The method according to claim 31, wherein said activity is a brute-force guessing of usernames in said database application.

Claim 33 (previously presented): The method according to claim 31, wherein said activity is the brute-force guessing of usernames and passwords for default accounts in said database application.

Appl. No. 10/798,079
Filed. March 11, 2004
Reply to Office action of September 11, 2006

Claim 34 (previously presented): The method according to claim 31, wherein said activity is the brute-force guessing of usernames and passwords for well-known accounts in said database application.

Claim 35 (previously presented): The method according to claim 31, wherein said activity is the scripting of password guessing against the database application.

Claim 36 (previously presented): A method for detecting suspicious activity in a database application, the method comprising:

monitoring for SQL statements executable in said database application which contain characteristics indicative of an attack;

actuating each batch statement in order to discover atomic SQL commands;

analyzing said atomic SQL commands against a pre-defined set of rules to identify said suspicious activity.

Claim 37 (previously presented): The method according to claim 36, wherein said suspicious activity is a use of comments within an SQL statement.

Claim 38 (previously presented): The method according to claim 36, wherein said suspicious activity is a use of a UNION keyword within an SQL statement.

Claim 39 (previously presented): The method according to claim 36, wherein said suspicious activity is a use of a keyword designed to suppress auditing data.

Appl. No. 10/798,079
Filed. March 11, 2004
Reply to Office action of September 11, 2006

Claim 40 (previously presented): A method for detecting use of keywords to suppress auditing of attacks in a database application, the method comprising:

- monitoring for SQL statements that contain a keyword, where said keyword results in audit data being suppressed;
- detecting a suppressed SQL statement;
- detecting a conclusion of said suppressed SQL statement;
- determining that no execution of said keyword designed to suppress said SQL statement actually occurred.

Claim 41 (previously presented): The method according to claim 40, further comprising a use of passwords designed to cause an auditing system to suppress text of said SQL statement and masking malicious activity.

Claim 42 (previously presented): A host-based intrusion prevention method for blocking attacks on database applications, the method comprising:

- detecting an attack occurring through a session with said database application;
- identifying a source of said attack;
- implementing a method of stopping said attack source;
- implementing a method of preventing further attacks from said attack source.

Claim 43 (previously presented): The method according to claim 42, wherein said method of stopping said attack source is killing a user connection of said attack source.

Appl. No. 10/798,079
Filed. March 11, 2004
Reply to Office action of September 11, 2006

Claim 44 (previously presented): The method according to claim 42, wherein said method of stopping said attack source is sending a reset to said attack source.

Claim 45 (previously presented): The method according to claim 42, wherein said method of stopping said attack source is blocking a SQL command.

Claim 46 (previously presented): The method according to claim 42, wherein said method of stopping said attack source is intercepting and filtering a SQL command.

Claim 47 (previously presented): The method according to claim 42, wherein said method of stopping said attack source is throwing an exception.

Claim 48 (previously presented): The method according to claim 42, wherein said method of preventing further attacks is disabling an account from being used.

Claim 49 (previously presented): The method according to claim 42, wherein said method of preventing further attacks is killing any future attempts from said attack source.

Claim 50 (previously presented): A method for detecting attempts to inject SQL into a database application, the method comprising:

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

monitoring for SQL statements executable in said database application and
intended to run queries not designed to be run by a middle-tier application;

analyzing said SQL statement's identifying characteristics indicative of SQL
injection;

implementing an action upon detection of identifying characteristics indicative of
SQL injection.

Claim 51 (previously presented): The method according to claim 50, wherein said
action is causing a security alert to be fired.

Claim 52 (previously presented): The method according to claim 50, wherein said
action is causing the SQL statement to be blocked.

Claim 53 (previously presented): A method for detecting attempts to inject SQL into
a database application, the method comprising:

listening to SQL queries executable on said database application for a determined
period of time;

tokenizing SQL statements into standard forms;

recording a combination and an order of tokens expected;

analyzing SQL statements received later to identify those that do not conform to
said expected combination of tokens.

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

Claim 54 (previously presented): A method for detecting malicious activity in a database application, the method comprising:

listening to SQL queries executable on said database application;

analyzing SQL statements by applying regular expressions to detect vulnerabilities;

sending alerts when an SQL statement matching a regular expression is discovered.

Claim 55 (previously presented): The method according to claim 54, wherein said regular expression is designed to detect a buffer overflow in a call from SQL to a built-in database function.

Claim 56 (previously presented): The method according to claim 54, wherein said regular expression is designed to detect a buffer overflow in a call from SQL to an operating system function.

Claim 57 (previously presented): The method according to claim 54, wherein said regular expression is designed to detect an attempt to escalate privileges of a user in said database application.

Claim 58 (previously presented): The method according to claim 54, wherein said regular expression is designed to detect an attempt to insert an SQL statement into a parameter of stored procedures.

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

Claim 59 (previously presented): The method according to claim 54, wherein said regular expression is designed to detect an attempt to escalate privileges of a user in an operating system.

Claim 60 (previously presented): A method for detecting activity which may result in cross-site scripting vulnerabilities, the method comprising:

- monitoring for SQL statements executable in said database application;
- actuating each batch statement in order to discover atomic SQL commands;
- examining an atomic SQL command for HTML tags.

Claim 61 (previously presented): The method according to claim 60, wherein said atomic SQL command contains an HTML tag.

Claim 62 (previously presented): The method according to claim 61, wherein said HTML tag is unencoded.

Claim 63 (previously presented): The method according to claim 61, wherein said HTML tag is hex encoded.

Claim 64 (previously presented): A method for monitoring all activity for security auditing, the method comprising:

- monitoring for an event generated by a database application;

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

actuating said event;

recording said event.

Claim 65 (previously presented): The method according to claim 64, wherein said event being generated comprises an SQL statement.

Claim 66 (previously presented): The method according to claim 64, wherein said event being generated comprises failed logins and successful logins.

Claim 67 (previously presented): The method according to claim 64, wherein said event being generated comprises incomplete attempts to access said database application.

Claim 68 (previously presented): The method according to claim 64, wherein said event being generated comprises DBA activity.

Claim 69 (previously presented): The method according to claim 64, wherein said event being generated comprises changes to a configuration.

Claim 70 (previously presented): The method according to claim 64, wherein said event being generated comprises enabling of application roles.

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

Claim 71 (previously presented): The method according to claim 64, wherein said event being generated comprises a method of granting, revoking, or denying permissions or privileges.

Claim 72 (previously presented): The method according to claim 64, wherein said event being generated comprises a utility event.

Claim 73 (previously presented): The method according to claim 72, wherein said utility event is a backup command.

Claim 74 (previously presented): The method according to claim 72, wherein said utility event is a restore command.

Claim 75 (previously presented): The method according to claim 72, wherein said utility event is a bulk insert command.

Claim 76 (previously presented): The method according to claim 72, wherein said utility event is a BCP command.

Claim 77 (previously presented): The method according to claim 72, wherein said utility event is a DBCC command.

Appl. No. 10/798,079

Filed. March 11, 2004

Reply to Office action of September 11, 2006

Claim 78 (previously presented): The method according to claim 64, wherein said event being generated comprises a server shutdown.

Claim 79 (previously presented): The method according to claim 64, wherein said event being generated comprises a pause.

Claim 80: (previously presented): The method according to claim 64, wherein said event being generated comprises a start-up.

Claim 81 (previously presented): The method according to claim 64, wherein said event being generated comprises an audit event.

Claim 82 (previously presented): The method according to claim 81, wherein said audit event is an add audit command.

Claim 83 (previously presented): The method according to claim 81, wherein said audit event is a modify audit command.

Claim 84 (previously presented): The method according to claim 81, wherein said audit event is a stop audit command.

Claim 85 (previously presented): The method according to claim 64, wherein said event being generated comprises use of extended stored procedures.

Appl. No. 10/798,079
Filed. March 11, 2004
Reply to Office action of September 11, 2006

Claim 86 (previously presented): A method for providing exceptions to security alerts, the method comprising:

monitoring for events generated by a database application;
filtering alerts raised that match a defined set of rules;
passing alerts not matching a normal definition of said defined set of rules.

Claim 87 (previously presented): The method according to claim 86, wherein said defined set of rules comprises values for each field collected for each event.

Claim 88 (previously presented): The method according to claim 86, wherein said filtering is matched by comparing values of each field with values defined in an exception.